P Paruta et al.

Simulation of plasma turbulence in the periphery of diverted tokamak by using the GBS code

Implementation of diverted magnetic equilibria in GBS, a numerical code to simulate plasma turbulence in the tokamak periphery

Paola Paruta, P. Ricci, F.Riva, C. Wersal, C. Beadle, B.Frei

August 3, 2018

Abstract

We report on the implementation of diverted magnetic equilibria in GBS, a simulation code used to evolve plasma turbulence in the tokamak periphery by solving the drift-reduced Braginskii's equations. The model equations are written in toroidal coordinates, abandoning flux coordinate systems, that are not defined at the X-point, and a fourth order finite difference scheme is used for the implementation of the spatial operators on poloidally and toroidally staggered grids. The GBS numerical implementation is verified through the method of manufactured solutions. The code convergence properties are tested.

1 Introduction

Plasma fueling, power exhaust, impurities and neutral dynamics are all governed by the complex physics that takes place in the plasma periphery of a tokamak device. Hence, understanding, predicting, and controlling the plasma turbulence in this region is of crucial importance for the success of fusion [1].

From a modelling point of view, multiple challenges arise when approaching the tokamak periphery. Phenomena occurring on a wide range of length and time scales are present, from the electron gyro-radius to the machine major radius; from the gyro-motion to the turbulence time scale. The presence of large amplitude turbulent structures does not allow for the decoupling between fluctuating and background quantities. In addition, a complex magnetic geometry characterises the tokamak periphery, as it is composed of the edge, where the magnetic field lines lie on closed flux surfaces, and the Scrape-Off Layer (SOL), where the magnetic field lines intersect the vessel wall.

GBS is a three-dimensional first-principles simulation code that has been developed in the past years in order to study the plasma turbulence dynamics in the tokamak periphery. Similarly to other codes developed for the same reason (BOUT++ [2], TOKAM3X [3], GBD [4], GRILLIX [5], HESEL [6]), GBS evolves the drift-reduced Braginskii's equations [7]-[8],

a set of fluid equations valid in the high-collisionality regime of the toka-mak periphery. In the past years GBS has contributed to progress our understanding of the SOL physics in limited configurations, where the SOL is defined by a toroidal or poloidal limiter. For example, GBS has provided predictions of the SOL width [9] in this configuration.

In the present work we extend GBS to the treatment of diverted toka-mak scenarios. In these configurations the magnetic field lines are diverted towards the vessel by using external field coils and the separation between the edge and the SOL is not defined by the limiter position, but rather by a flux surface, called the separatrix, that intersects itself at one or mul-tiples points denoted as X-points. From a modelling point of view, the main additional challenge that diverted scenarios bring is the choice of a proper coordinate system.

As charged particles move freely along the magnetic field line, whereas their motion is constrained in the direction orthogonal to it, parallel and perpendicular dynamics in a magnetised plasma occur on very different length scales. In particular, most plasma properties are approximately constant on flux surfaces and vary sharply across them. Therefore a coor-dinate system where the basis vectors identify parallel and perpendicular directions has the advantage of reducing the computational cost of the simulations. In most code implementations, as well as in the limited ver-sion of GBS [10], a coordinate whose contour lines correspond to the flux surfaces is chosen as one of the two perpendicular coordinates. Such flux coordinate can be identified with the magnetic poloidal flux ψ, the func-tion that allows expressing the tokamak toroidally symmetric equilibrium magnetic field as:

$$\mathbf{B} = F(\psi)\nabla\varphi + \nabla\psi \times \nabla\varphi \tag{1}$$

which ensures $\nabla \cdot \mathbf{B} = 0$ [11]. The first term on the right-hand side of eq. (1) represents the toroidal component of the magnetic field, $F(\psi)$ be-ing a scalar function of ψ, and the second term identifies the poloidal component, φ being the toroidal angle. For a toroidally symmetric mag-netic field, the poloidal flux ψ does not depend on φ, $\nabla\varphi \cdot \nabla\psi = 0$. From eq. (1), it follows that $\mathbf{B} \cdot \nabla\psi = 0$. This shows that ψ contour lines cor-respond indeed to the flux surfaces on which the magnetic field lies and that the contravariant basis vector $\nabla\psi$ always conveniently points in a direction orthogonal to the magnetic field line and to the flux surface, as desired.

In diverted equilibria, the use of a flux coordinate is complicated by the presence of one or multiple X-points. At these locations, the magnetic field is purely toroidal and has no component in the poloidal plane. Equa-tion (1) implies $\nabla\psi \times \nabla\varphi = 0$ and consequently $\nabla\psi = 0$ at the X-point. In fact, the X-point is defined as a saddle point of the magnetic flux ψ, i.e. $\nabla\psi\big|_{\mathrm{Xpt}} = 0$. As a consequence, the Jacobian of a system that uses the flux coordinate is singular at the X-point, $J = (\nabla\psi \cdot \nabla\alpha \times \nabla\varphi)^{-1} = \infty$ ($\nabla\alpha$ denoting a third basis vector, in addition to $\nabla\psi$ and $\nabla\varphi$). Moreover, the use of a flux coordinate is made problematic by the presence of the flux expansion around the X-point. In fact, in the proximity of the X-point, since $\nabla\psi$ is small, the spacing between the ψ contour lines becomes larger than at the midplane. This makes it challenging from a numerical stand-

point to attain a good physical resolution around the X-point. Indeed, a uniform spacing in ψ does not correspond to a uniform spacing in physical space, and an over-resolution at mid-plane is needed in order to attain a good resolution around the X-point, increasing the computational cost of a simulation.

Various approaches are being followed to tackle the singularity of the flux coordinate in diverted equilibria. In BOUT++ [2] and TOKAM3X [3], the use of the flux coordinate is retained also in diverted scenarios, but the numerical grid points are generated to avoid falling on the X-point position. This bypasses the problem of the singularity at the X-point but does not solve the low resolution issue due to the flux expansion. In GRILLIX [5], flux coordinates are abandoned in favour of Cartesian coordinates in the poloidal plane. To compensate the lack of alignment of the coordinates to the magnetic flux, an effort to accurately capture the parallel direction is put in place by carefully handling the parallel operator with the Flux Coordinate Independent (FCI) method [12]-[13].

In GBS we choose to step away from the use of flux coordinates and, instead, use the toroidal coordinates (r, θ, φ), which are defined as:

$$
\begin{aligned}
x &= R \cos \varphi = (R_0 - r \cos \theta) \cos \varphi \\
y &= R \sin \varphi = (R_0 - r \cos \theta) \sin \varphi \\
z &= Z_0 + r \sin \theta
\end{aligned}
\tag{2}
$$

being R the distance from the symmetry axis of the torus, θ the poloidal angle, and r the distance in the poloidal plane from the point located at $R = R_0$ and vertical position Z_0.

This allows us to easily enclose the edge and SOL of a diverted plasma in a domain that corresponds to a rectangular box in the (r, θ, φ) coordinates, unlike in the case of flux coordinates or Cartesian coordinates. In addition, the coordinate system and numerical method we propose have the advantage of being flexible, allowing the straightforward implementation of double-null [14] or snowflakes [15] configurations. To compensate for the loss of alignment of the coordinate system to the magnetic field, we increase the order of accuracy of the numerical scheme (from second to fourth order finite differences) and use grid staggering.

The present paper is organised as follows. Section 2 describes the physical model we consider to study the tokamak periphery. Section 3 focuses on the numerical implementation of diverted configurations in GBS. Special attention is given to the implementation of the boundary conditions. In section 4 the implementation of the numerical algorithm in GBS is verified by using the method of manufactured solutions. Results on code convergence are reported in section 5. The conclusions follow.

2 Physical model

The present section details the model used by GBS to evolve plasma turbulence in the tokamak periphery. The differential operators present in the model are specified for the toroidal coordinates we use (their thorough analytical derivation is carried out in the Appendix). The boundary and initial conditions are described in detail.

2.1 Drift-reduced Braginskii's equations

Since the plasma at the periphery of a tokamak device is sufficiently collisional that deviations from a Maxwellian distribution are small, a fluid description, such as the one derived by Braginskii [7], is generally used. In addition, turbulence occurs on a time scale considerably longer than the gyro-motion ($\partial_t \ll \Omega_{ci} = eB/(m_i c)$) and on a length scale larger than the ion sonic gyro-radius, $\rho_s = c_s/\Omega_{ci}$, with $c_s = \sqrt{T_e/m_i}$. The drift approximation of the particle orbits can therefore be used to simplify Braginskii's equations, as shown by Zeiler [8].

We note that the drift-reduced Braginskii's equations were first implemented in the GBS code in limited configuration in the electrostatic, cold-ion limit, assuming no interaction of the plasma with the neutrals, as described in Ref. [16]. In the following years, the hot ion physics was introduced by Mosetto *et al.* [17], electromagnetic effects by Halpern [18] and coupling with the neutral dynamics by Wersal *et al.* [19]. The most complete model used for limited simulations is summarised in the paper by Halpern *et al.* [20]. The version of the drift-reduced Braginskii's equations considered in the present paper does not include coupling with neutrals, nor electromagnetic effects, but does include hot ions. The Boussinesq approximation in the evaluation of the divergence of the polarisation current is used to reduce the cost. Finally, we assume $\nabla \cdot \mathbf{b} \simeq 0$ when computing the parallel advection terms. The model equations we consider are therefore the following:

$$\frac{\partial n}{\partial t} = -\frac{\rho_\star^{-1}}{B}[\phi, n] + \frac{2}{B}\left[C(p_e) - nC(\phi)\right] - \nabla_\parallel(nv_{\parallel e}) + S_n + D_n\nabla_\perp^2 n \tag{3}$$

$$\frac{\partial v_{\parallel e}}{\partial t} = -\frac{\rho_\star^{-1}}{B}[\phi, v_{\parallel e}] - v_{\parallel e}\nabla_\parallel v_{\parallel e} \tag{4}$$
$$+\frac{m_i}{m_e}\left(\nu j_\parallel + \nabla_\parallel\phi - \frac{1}{n}\nabla_\parallel p_e - 0.71\nabla_\parallel T_e\right)$$
$$-\frac{2}{3n}\frac{m_i}{m_e}\nabla_\parallel G_e + D_{v_{\parallel e}}\nabla_\perp^2 v_{\parallel e}$$

$$\frac{\partial v_{\parallel i}}{\partial t} = -\frac{\rho_\star^{-1}}{B}[\phi, v_{\parallel i}] - v_{\parallel i}\nabla_\parallel v_{\parallel i} - \frac{1}{n}\nabla_\parallel(p_e + \tau p_i) \tag{5}$$
$$-\frac{2}{3n}\nabla_\parallel G_i + D_{v_{\parallel i}}\nabla_\perp^2 v_{\parallel i}$$

$$\frac{\partial T_e}{\partial t} = -\frac{\rho_\star^{-1}}{B}[\phi, T_e] - v_{\parallel e}\nabla_\parallel T_e + \frac{4}{3}\frac{T_e}{B}\left[\frac{1}{n}C(p_e) + \frac{5}{2}C(T_e) - C(\phi)\right] \tag{6}$$
$$+\frac{2}{3}T_e\left[0.71\nabla_\parallel v_{\parallel i} - 1.71\nabla_\parallel v_{\parallel e} + 0.71(v_{\parallel i} - v_{\parallel e})\frac{\nabla_\parallel n}{n}\right]$$
$$+S_{T_e} + \chi_{\perp,e}\nabla_\perp^2 T_e + \chi_{\parallel,e}\nabla_\parallel^2 T_e$$

$$\frac{\partial T_i}{\partial t} = -\frac{\rho_\star^{-1}}{B}[\phi, T_i] - v_{\parallel i}\nabla_\parallel T_i + \frac{4}{3}\frac{T_i}{B}\left[C(T_e) + \frac{T_e}{n}C(n) - C(\phi)\right] \tag{7}$$
$$+\frac{2}{3}T_i\left(v_{\parallel i} - v_{\parallel e}\right)\frac{\nabla_\parallel n}{n} - \frac{2}{3}T_i\nabla_\parallel v_{\parallel e} - \frac{10}{3}\tau\frac{T_i}{B}C(T_i) + S_{T_i}$$

$$\frac{\partial \omega}{\partial t} = -\frac{\rho_\star^{-1}}{B}[\phi, \omega] - v_{\parallel i}\nabla_\parallel\omega + \frac{B^2}{n}\nabla_\parallel j_\parallel + \frac{2B}{n}C(p_e + \tau p_i) \tag{8}$$
$$+\frac{B}{3n}C(G_i) + D_\omega\nabla_\perp^2\omega$$

$$\nabla_\perp^2\phi = \omega - \tau\nabla_\perp^2 T_i \tag{9}$$

In eqs. (3)-(9) all variables are dimensionless (in the following, we use a tilde to denote physical variables, unless specified otherwise). We define the plasma density $n = \tilde{n}/n_0$, the electron temperature $T_e = \tilde{T}_e/T_{e0}$, the ion temperature $T_i = \tilde{T}_i/T_{i0}$, the electro-static potential $\phi = e\tilde{\phi}/T_{e0}$, the electron parallel velocity $v_{\parallel e} = \tilde{v}_{\parallel e}/c_{s0}$, the ion parallel velocity $v_{\parallel i} = \tilde{v}_{\parallel i}/c_{s0}$ and the vorticity $\omega = \tilde{\omega}\, e\rho_{s0}^2/T_{e0}$ with $n_0, T_{e0}, T_{i0}, c_{s0} = \sqrt{T_{e0}/m_i}$ and $\rho_{s0} = c_{s0}/\Omega_{ci}$ reference density, temperatures, sound velocity and ion sonic Larmor radius expressed in physical units. The electron and ion pressures are denoted as $p_e = nT_e$ and $p_i = nT_i$. The dimensionless current is $j_\parallel = n(v_{\parallel i} - v_{\parallel e})$. Time is defined as $t = \tilde{t}\, c_{s0}/R_0$, where R_0 is the major radius at magnetic axis, in physical units. The dimensionless parameters appearing in the model equations are: $\rho_\star = \rho_{s0}/R_0$ (normalised ion sonic Larmor radius), $\nu = en_{e0}R_0/(m_i c_{s0}\sigma_i)$ (normalised Spitzer resistivity), $\tau = T_{i0}/T_{e0}$ (ion to electron temperature ratio). In the density and temperature equations S_n and $S_{T_{e,i}}$ denote source terms that mimic the outflow of plasma and heat from the core. The gyroviscous terms $G_{e,i}$

are defined as:

$$G_e = -\eta_{0,e} \left(2\nabla_\parallel v_{\parallel e} - \frac{C(p_e)}{Bn} + \frac{C(\phi)}{B} \right) \tag{10}$$

$$G_i = -\eta_{0,i} \left(2\nabla_\parallel v_{\parallel i} + \tau \frac{C(p_i)}{Bn} + \frac{C(\phi)}{B} \right) \tag{11}$$

with $\eta_{0,e,i}$ constant coefficients. Small numerical diffusion terms of the type $D_f \nabla_\perp^2 f$ are added for numerical stability.

2.2 Differential operators in toroidal coordinates

The dimensionless spatial operators present in eqs. (3)-(9) are the parallel gradient, $\nabla_\parallel$, the parallel diffusion operator $\nabla_\parallel^2$, the Poisson brackets $[\phi, f]$, the curvature operator C, and the perpendicular diffusion operator $\nabla_\perp^2$:

$$\nabla_\parallel = R_0 \mathbf{b} \cdot \tilde{\nabla} \tag{12}$$

$$\nabla_\parallel^2 = R_0^2 \mathbf{b} \cdot \tilde{\nabla}(\mathbf{b} \cdot \tilde{\nabla}) \tag{13}$$

$$[\phi, f] = \rho_{s0}^2 \mathbf{b} \cdot (\tilde{\nabla}\phi \times \tilde{\nabla}f) \tag{14}$$

$$C = R_0 \rho_{s0} \frac{\tilde{B}}{2} (\tilde{\nabla} \times \frac{\mathbf{b}}{\tilde{B}}) \cdot \tilde{\nabla} \tag{15}$$

$$\nabla_\perp^2 = \rho_{s0}^2 \tilde{\nabla} \cdot ((\mathbf{b} \times \tilde{\nabla}) \times \mathbf{b}) \tag{16}$$

They depend on the norm and on the versor of the magnetic field, $\tilde{B}$ and $\mathbf{b} = \tilde{\mathbf{B}}/\tilde{B}$.

As shown in detail in Appendix A, these spatial operators can be expanded in terms of the parameters $\varepsilon = a/R_0$, $\sigma = l_p/R_0$, and $\delta = \rho_{s0}/R_0$, where a is the tokamak minor radius and l_p is the gradient length scale in the poloidal direction for the plasma fluctuating quantities, in physical units. Herein, we neglect all terms that are first order or higher in these parameters. Neglecting order ε terms corresponds to considering the large aspect ratio approximation, while neglecting order σ and δ terms is justified because l_p and ρ_{s0} are much smaller than the machine major radius R_0, that also identifies the typical gradient scale length in the toroidal direction.

In these limits and assuming that the toroidal magnetic field is inversely proportional to the major radius, i.e. $F(\psi) = B_0 R_0$ and $\tilde{\mathbf{B}}_{tor} = B_0 R_0 \nabla\varphi$ (see eq. (1)), the operators in eqs. (3)-(9) take the following form in the (r, θ, φ) toroidal coordinates:

$$\nabla_\parallel f = \frac{B_0}{|B_0|} \frac{\partial f}{\partial \varphi} + \partial_r \psi \frac{1}{r} \frac{\partial f}{\partial \theta} - \frac{1}{r} \partial_\theta \psi \frac{\partial f}{\partial r} \tag{17}$$

$$\nabla_\parallel^2 f = \nabla_\parallel(\nabla_\parallel f) \tag{18}$$

$$[\phi, f] = \frac{1}{r} \frac{B_0}{|B_0|} \left(\frac{\partial \phi}{\partial r} \frac{\partial f}{\partial \theta} - \frac{\partial \phi}{\partial \theta} \frac{\partial f}{\partial r} \right) \tag{19}$$

$$C(f) = \frac{B_0}{|B_0|} \left(\sin\theta \frac{\partial f}{\partial r} + \frac{\cos\theta}{r} \frac{\partial f}{\partial \theta} \right) \tag{20}$$

$$\nabla_\perp^2 f = \frac{\partial^2 f}{\partial r^2} + \frac{1}{r^2} \frac{\partial^2 f}{\partial \theta^2} \tag{21}$$
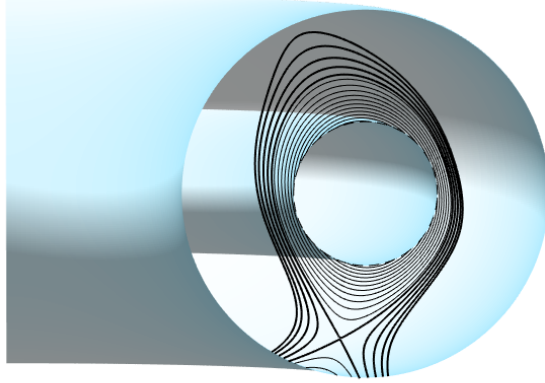
Figure 1: Sketch of the computational domain of GBS with a contour plot of the poloidal flux ψ in the edge (thin lines) and in the SOL (thick lines).

where $r = \tilde{r}/\rho_{s0}$. Here we are using the dimensionless poloidal flux $\psi = \tilde{\psi}/(\rho_{s0}^2 |B_0|)$. We note that the perpendicular operators in eqs (19)-(21) contains only derivatives in the poloidal plane, considerably simplifying their numerical implementation. Finally, we would like to stress the flexibility resulting from the use of toroidal coordinates. All expressions of the geometrical operators are reduced to combinations of derivatives in (r, θ, φ) multiplied by coefficients that depend only on the equilibrium magnetic field. These coefficients are computed only once at the beginning of the simulation. The resulting scheme is very flexible, since by prescribing these values as input parameter, any axisymmetric magnetic equilibrium can be investigated: single null, double null or snowflake.

2.3 Radial boundary conditions

The domain where eqs. (3)-(9) are solved is a toroidal ring, $r_{min} \leq r \leq r_{max}$, $0 \leq \theta < 2\pi$, $0 \leq \varphi < 2\pi$. The ring is centered at the tokamak magnetic axis and contains a closed flux surface region, the separatrix, the X-point, and divertor legs (see fig. 1).

The magnetic field lines cross the boundary towards the core, $r = r_{min}$, and the boundary at the vessel wall, $r = r_{max}$, since these are generally not aligned to the flux surfaces. At $r = r_{max}$, the plasma interaction with the solid wall is described by the magnetic pre-sheath boundary conditions developed by Loizu *et al.* [21] in the cold ion limit and then extended by

Mosetto *et al.* [17] to include hot ion effects:

$$v_{\parallel,i} = \pm\sqrt{T_e}F_T$$

$$v_{\parallel,e} = \pm\sqrt{T_e}\max\{\exp(\Lambda - \frac{\phi}{T_e}), \exp(\Lambda)\}$$

$$\partial_r\phi = \mp\frac{\sqrt{T_e}}{F_T}\partial_r v_{\parallel,i}$$

$$\partial_r n = \mp\frac{n}{\sqrt{T_e}F_T}\partial_r v_{\parallel,i} \qquad (22)$$

$$\omega = -\frac{1}{F_T^2}(\partial_r v_{\parallel,i})^2 \mp \frac{\sqrt{T_e}}{F_T}\partial_{rr}^2 v_{\parallel,i}$$

$$\partial_r T_e = 0$$

$$\partial_r T_i = 0$$

with $F_T = \sqrt{1 + \tau T_i/T_e}$. The plus/minus indicates whether the magnetic field points towards (top sign) or out from the wall (bottom sign). Note that, for simplicity, the boundary conditions in eqs. (22) neglect the terms containing derivatives along the wall, included in ref. [21].

At the core boundary, $r = r_{\min}$, we use an *ad hoc* set of boundary conditions, i.e. $\partial_r f = 0$ for all fields f except for ω and ϕ, for which we impose $\omega = 0$ and $\phi = \Lambda T_e$. GBS mimics the outflow of plasma from the core through a source of plasma density and temperature ($S_n, S_{T_e,i}$ terms in eqs. (3),(6) and (7)), radially localised in the closed flux region. The region that extends from the inner radial boundary to the source location is a buffer volume, which is not subject of physics investigations. The boundary conditions are periodic in the poloidal and toroidal directions.

2.4   Initial conditions

The long term evolution of the system is statistically independent of the initial conditions. For this reason the initial conditions are not physically interesting, and they are just chosen compatible with the boundary conditions. We impose $\phi(r, \theta, \varphi) = \Lambda T_e$, with T_e being an initial constant electron temperature. Similarly, ω, n and T_i are set to a constant value on the entire domain. Additionally, we impose $v_{\parallel e} = v_{\parallel i}$, with the $v_{\parallel e}, v_{\parallel i}$ functions satisfying $v_{\parallel e,i}|_{r_{\max}} = \pm\sqrt{T_e}$ and $\partial_r v_{\parallel e,i}|_{r_{\max}} = 0$, such that the right-hand side of the boundary conditions for n, ϕ, ω is zero at the wall (see eqs. (22)), according to uniform initial profiles of these quantities. Finally, for all fields, the initial conditions present no toroidal dependence, except for a three-dimensional random noise that is added to seed plasma turbulence.

3   Numerical implementation

The new numerical implementation of the model described in section 2.1 is largely similar to the GBS code for limited plasma simulations [20], which is based on a fourth order Runge-Kutta time stepping and second

order centered finite differences for the spatial discretisation of the operators. With respect to the limited version, we implement the differential operators in toroidal rather than flux coordinates and at fourth order finite differences rather than second. Additionally, the magnetic pre-sheath boundary conditions are now applied all around the outer radial boundary rather than at the limiter. In the following we describe the main features of the numerical implementation of diverted geometries in GBS, highlighting the differences with respect to the limited version.

3.1 Spatial discretisation

We discretise our domain by using a uniform numerical grid of N_r, N_θ, and N_φ points along the radial, poloidal and toroidal directions respectively. We stagger the grid that discretises the n, ϕ, ω, T_e and T_i fields (n-grid) with respect to the grid where we evaluate $v_{\parallel e}$ and $v_{\parallel i}$ (v-grid) in the toroidal and poloidal directions. Staggered-grids were first used by Harlow and Welch in 1965 [22] to provide a remedy to the checkerboard patterns that can appear when treating an advection problem with centered finite difference, as shown in [23]. The idea behind grid staggering can be shown by considering a minimal system, contained in the drift-reduced Braginskii's equations (3) and (4), that describes the evolution of density and electron parallel velocity, i.e.:

$$\partial_t N + \nabla_\parallel v_{\parallel e} = 0$$
$$\partial_t v_{\parallel e} + T_e \nabla_\parallel N = 0$$

where we indicate $N = \log(n)$ and we consider T_e constant for simplicity. If a second order centered finite difference scheme is used to discretise the parallel derivative of N and $v_{\parallel e}$ on the same uniform grid of spacing Δx, together with an explicit Euler time discretisation with step Δt, we obtain:

$$N_j^{m+1} = N_j^m - \frac{\Delta t}{2\Delta x} \left(v_{\parallel e, j+1}^m - v_{\parallel e, j-1}^m \right) \tag{23}$$

$$v_{\parallel e, j}^m = v_{\parallel e, j}^{m-1} - \frac{\Delta t}{2\Delta x} T_e \left(N_{j+1}^{m-1} - N_{j-1}^{m-1} \right) \tag{24}$$

Here j and m indicate the spatial and temporal grid index, i.e. $N_j^m = N(x_j, t_m)$. Combining equations (23) and (24) we derive:

$$\frac{1}{\Delta t^2} \left(N_j^{m+1} - 2N_j^m + N_j^{m-1} \right) = \frac{1}{4\Delta x^2} T_e \left(N_{j-2}^{m-1} - 2N_j^{m-1} + N_{j+2}^{m-1} \right) \tag{25}$$

We note that in eq. (25) N_j^{m+1} only depends on values of N at the $j-2$, j and $j+2$ points. The values of N on even and odd grid points are therefore decoupled. This decoupling allows checkerboard patterns, i.e. a solution with a, b such that $N_j = a$ for even j, and $N_j = b$ for odd j, with $a \neq b$. Shifting the position of the grid point at which v is evaluated by $\Delta x/2$ midway between two n-grid points, i.e. "staggering" the n and v-grids, avoids the formation of a checkerboard pattern.

Jolliet *et al.* [24] carried out an initial investigation of possible numerical implementations for the parallel gradient operator in GBS, showing

the good properties of fourth order centered finite difference in limited scenarios. Leveraging the results of this work, we proceed with the implementation of fourth order finite difference operators in GBS. We describe the discretisation of derivatives in one-dimension, since all GBS operators, except for the Poisson brackets, are computed as a linear combination of derivatives in r, θ, and φ. More precisely, the differential operators can be written in terms of the discretised first derivatives $D_x, D_x^{n2v}, D_x^{v2n}$, the discretised second derivative D_{xx}, and of the interpolation between staggered grids, I_x^{n2v}, I_x^{v2n}, which are defined as:

$$D_x f_j = \frac{1}{\Delta x} \left[\frac{1}{12} f_{j-2} - \frac{2}{3} f_{j-1} + \frac{2}{3} f_{j+1} - \frac{1}{12} f_{j+2} \right] \tag{26}$$

$$D_x^{n2v} f_j = \frac{1}{\Delta x} \left[\frac{1}{24} f_{j-2} - \frac{9}{8} f_{j-1} + \frac{9}{8} f_j \quad - \frac{1}{24} f_{j+1} \right] \tag{27}$$

$$D_x^{v2n} f_j = \frac{1}{\Delta x} \left[\frac{1}{24} f_{j-1} - \frac{9}{8} f_j \quad + \frac{9}{8} f_{j+1} - \frac{1}{24} f_{j+2} \right] \tag{28}$$

$$D_{xx} f_j = \frac{1}{\Delta x^2} \left[-\frac{1}{12} f_{j-2} + \frac{4}{3} f_{j-1} - \frac{5}{2} f_j + \frac{4}{3} f_{j+1} - \frac{1}{12} f_{j+2} \right] \tag{29}$$

$$I_x^{n2v} f_j = \left[-\frac{1}{16} f_{j-2} - \frac{9}{16} f_{j-1} + \frac{9}{16} f_j \quad - \frac{1}{16} f_{j+1} \right] \tag{30}$$

$$I_x^{v2n} f_j = \left[-\frac{1}{16} f_{j-1} - \frac{9}{16} f_j \quad + \frac{9}{16} f_{j+1} - \frac{1}{16} f_{j+2} \right] \tag{31}$$

where x stands for one of the three coordinates r, θ, or φ, and the apex $n2v$ ($v2n$) indicates that the input field is defined on the n-grid (v-grid) and the output on a v-grid (n-grid) (see fig. 2). For example, the advection term in the density equation (3), evaluated on the n-grid point $(r_j, \theta_k, \varphi_l)$, is computed as:

$$\left(n \nabla_\parallel v_{\parallel e} \right)_{j,k,l} = n \left(c_1 \frac{\partial v_{\parallel e}}{\partial \varphi} + c_2 \frac{\partial v_{\parallel e}}{\partial \theta} + c_3 \frac{\partial v_{\parallel e}}{\partial r} \right)$$
$$\simeq n_{i,j,k} \left(c_1 I_\theta^{v2n} D_\varphi^{v2n} v_{\parallel e} + c_2 I_\varphi^{v2n} D_\theta^{v2n} v_{\parallel e} + c_3 I_\theta^{v2n} I_\varphi^{v2n} D_r v_{\parallel e} \right)_{j,k,l}$$

where $c_1 = B_0/|B_0|$, $c_2 = \partial_{\bar{r}} \psi a / \rho_{s0}$, $c_3 = -\partial_\theta \psi a / \rho_{s0}$, from eq. (17). Note that the interpolation is performed only along the θ and φ directions, since there is no staggering in r. We also note that all the above operators require a 5-point stencil $[j-2, j+2]$ and that $n2v$ and $v2n$ operators use the same coefficients.

Two additional operators, D_{xx}^{n2v} and D_{xx}^{v2n}, are needed for the curvature-related contribution to the gyroviscous terms, in eqs. (10)-(11), i.e. $C(\nabla_\parallel(f))$, $\nabla_\parallel(C(f))$ and $C(C(f))$. Since a fourth order implementation of these operators requires a 7-points stencil, which impacts the number of ghost points in the treatment of the MPI subdomain boundaries (see Sec. 3.3), D_{xx}^{n2v} and D_{xx}^{v2n} are implemented at second order:

$$D_{xx}^{n2v} f_j = \frac{1}{\Delta x^2} \left[\frac{1}{2} f_{j-2} - \frac{1}{2} f_{j-1} - \frac{1}{2} f_j \quad + \frac{1}{2} f_{j+1} \right] \tag{32}$$

$$D_{xx}^{v2n} f_j = \frac{1}{\Delta x^2} \left[\frac{1}{2} f_{j-1} - \frac{1}{2} f_j \quad - \frac{1}{2} f_{j+1} + \frac{1}{2} f_{j+2} \right] \tag{33}$$
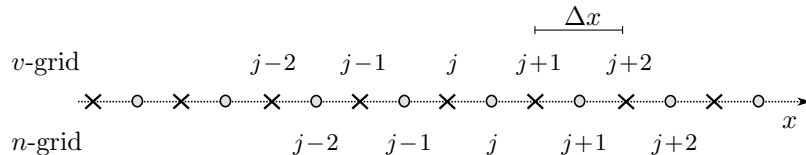
$$\overset{\Delta x}{\longmapsto}$$

v-grid $\qquad\qquad j{-}2 \quad j{-}1 \quad j \quad j{+}1 \quad j{+}2$

$\times\!\cdots\!\circ\!\cdots\!\times\!\cdots\!\circ\!\cdots\!\times\!\cdots\!\circ\!\cdots\!\times\!\cdots\!\circ\!\cdots\!\times\!\cdots\!\circ\!\cdots\!\times\!\cdots\!\circ\!\cdots\!\times\!\cdots\!\circ\!\longrightarrow$

$\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad x$

n-grid $\qquad\qquad\quad\; j{-}2 \quad j{-}1 \quad j \quad\; j{+}1 \quad j{+}2$

Figure 2: Sketch of the grid staggering performed in the θ and φ. Here x is either one of these directions. The velocities $v_{\parallel e}$ and $v_{\parallel i}$ are evaluated on v-grid points, indicated with crosses, while n, ω, ϕ, T_e and T_i are evaluated on n-grid points, indicated with circles. The labeling of the grid points is useful to interprete the expressions of the discretised derivatives and discrete interpolation in eq. (26)-(31), that allow operating between the two grids.

In the limited version of GBS, the Poisson brackets are discretised by using the Arakawa scheme at second order [25]. We keep the use of Arakawa scheme in the diverted version but we implement it at fourth order [26].

Finally, to compute the electric potential according to eq. (9), one needs to invert the perpendicular diffusion operator $\nabla_\perp^2 = \partial_{rr}^2 + 1/r^2\partial_{\theta\theta}^2$, see eq. (21). This is done by using a LU factorisation of the matrix resulting from the fourth order discretisation of this operator, computed once for all at the beginning of the simulation.

3.2 Boundary conditions

Two ghosts points are added on each side of the radial domain, i.e. $r = r_{min}$ and $r = r_{max}$, to impose Dirichelet and Neumann boundary conditions. By indicating the grid points inside the domain with indices $i = 1, ..., N_r$, the four ghosts points have indices $i = -1, 0$ at the boundary $r = r_{min}$ and $i = N_r + 1$, $N_r + 2$ at the vessel wall. We impose that the boundary conditions are satisfied midway between the $i = 0$ and $i = 1$ points and between the $i = N_r$ and $i = N_{r+1}$ points, for core and wall boundaries respectively. In practice, to implement the Dirichelet boundary condition $f(r_{min}) = f_b$ for ω and ϕ, we impose $(f_0 + f_1)/2 = f_b$ and, at the same time, $(f_{-1} + f_2)/2 = f_b$. The same scheme is applied to ω to impose the Dirichelet boundary conditions at the wall $r = r_{max}$. On the other hand, to impose the Dirichelet conditions at $r = r_{max}$ for $v_{\parallel e}$ and $v_{\parallel i}$ we impose $f_{N_r+2} = f_{N_r+1} = f_b$. To implement Neumann boundary condition $\partial_r n|_{r_{min}} = f_b$ (and similarly for $T_e, T_i, v_{\parallel e}, v_{\parallel i}$), we set $(f_1 - f_0)/\Delta r = f_b$ and $(f_0 - f_{-1})/\Delta r = f_b$. The same holds at $r = r_{max}$ for $n, T_e, T_i,$ and ϕ. These schemes are preferred to a fourth-order algorithm for numerical stability.

A discontinuity arises in the velocities boundary condition at the locations where the magnetic field is tangent to the wall, see eq. (22). At these locations, the boundary condition for parallel ion velocity presents a jump from $-\sqrt{T_e}$ to $+\sqrt{T_e}$, and a similar discontinuity arises for $v_{\parallel e}$. This issue is solved by applying a smoothing function from $+\sqrt{T_e}$ to $-\sqrt{T_e}$ so the $v_{\parallel i}$ boundary condition varies continuously at the wall. The discontinuity

11

present in the $v_{\parallel e}$ boundary condition is treated similarly. The discontinuity in n and ϕ boundary is only apparent, as the term $\mp \partial_r v_{\parallel i}$ remains generally negative across the points where **B** is tangent to the wall. We remark that this *ad hoc* smoothing function is required since a rigorous derivation of the magnetic pre-sheath boundary conditions for **B** tangent to the wall has not yet been developed, despite recent significant work on the subject [27]-[28].

3.3 Parallelisation

The use of the fourth order Runge-Kutta explicit time stepping method allows GBS to be easily parallelised. Domain decomposition is performed in all three coordinates (r, θ, φ) and ghost cell passing is carried out by using standard MPI calls. We note that the use of 5-points stencils for the numerical operators (see section 3.1) requires two ghosts points to be passed in each direction.

For the computation of the electric potential ϕ, eq. (9), a direct solver based on the MUMPS library [29]-[30] is used. An iterative multigrid method is also implemented in GBS to allow for a massive parallelisation of the solution of the Laplace operator in the poloidal plane [20], but it is only available for the second order finite difference scheme in the limited scenario at the moment (an ongoing effort is targeted to port the multigrid solver to fourth order).

Scalability tests of the new version of the GBS code are performed using the CPU partition of the Piz Daint supercomputer (hybrid Cray XC40) at the Swiss National Supercomputing Center in Lugano, Switzerland. Figure 3 shows the results of a strong scaling test (left), where the grid size is kept constant while the number of cores is increased, and of a weak scaling test (right), where the grid size and the number of cores in φ are increased simultaneously keeping their ratio constant. For both scalings the inverse normalised elapsed times, t_{Ncores}/t_{N_0}, i.e. the *speedup* for the strong scaling and the *efficiency* for the weak scaling, are plotted as a function of the number of cores ($Ncores$). Good scaling properties are observed up to 4608 cores for a grid of $N_r \times N_\theta \times N_\varphi = 512 \times 1024 \times 256$, which corresponds to that of a simulation of a medium size tokamak.
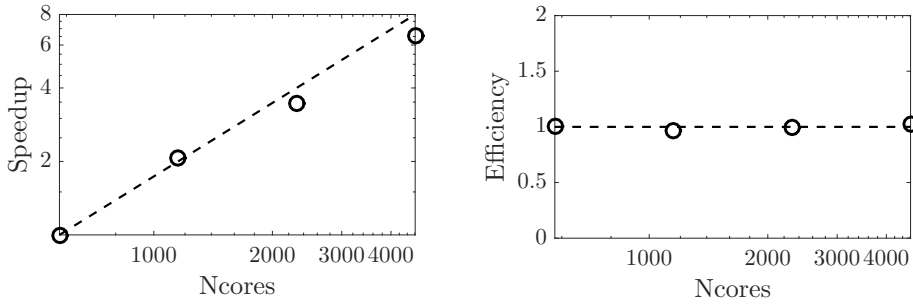
Figure 3: Results of strong (left) and weak (right) scaling tests on Piz Daint. In the strong scaling the grid size is fixed to $N_r \times N_\theta \times N_\varphi = 512 \times 1024 \times 256$ and the number of cores is increased. The speedup, i.e. the inverse of the normalised elapsed time, is expected to scale linearly with the number of cores in the ideal case. In the weak scaling, the ratio of grid size to number of cores stays constant as the number of cores increases ($N_r \times N_\theta \times N_\varphi = 512 \times 1024 \times N_\varphi$). In this case the efficiency, i.e. the normalised elapsed time, is expected to be constant.

4 Verification of GBS with the method of manufactured solution

We verify the new version of the GBS code using the method of manufactured solutions (MMS), a technique widely adopted by the computational fluid dynamics community [31] and first applied to fully verify a plasma turbulent code by Riva *et al.* [32] for GBS in limited magnetic configuration. We remark that the objective of the MMS is to verify that the discretised model equations have been implemented correctly in the code, not to validate the choice of the physical model. Herein we briefly present the basic idea behind the MMS and refer to Ref. [32] for a more detailed description of this methodology.

Given a model M with s its analytical solution (i.e. $M(s) = 0$), we aim at testing the implementation of a numerical discretisation of M, denoted as M_h, with h the discretisation parameter, through estimate of the error $e_h = \|s - s_h\|$, where s_h is the numerical solution of M_h (i.e. $M_h(s_h) = 0$). Since s is unknown, e_h cannot be evaluated. However, one can choose an arbitrary function u, referred to as the manufactured solution, compute the source term $S = M(u)$ analytically, solve $M_h(u_h) - S = 0$ numerically, and study $\tilde{e}_h = \|u - u_h\|$. Since the source term S is exact, the error $\tilde{e}_h$ is due to the discretisation of M and in our case it is expected to decrease as h^4 when as $h \to 0$ since we use fourth order discretisation schemes (both in space and time). In practice, one needs to compute

$$p = \frac{\ln(\tilde{e}_{rh}/\tilde{e}_h)}{\ln(r)} \tag{34}$$

where rh indicates the coarsening of the temporal and spatial mesh by a factor r, and show that $p \to 4$ for $h \to 0$.

13

In order to carry out the GBS code verification, we consider the diverted flux function plotted in fig. 4:

$$\psi(\bar{r}, \theta) = k(2u^3 - 2u^2 - (3/2 + \cos\theta)u + 1) \qquad (35)$$

where $u = (r - a/\rho_{s0})/(r_{max} - r_{min})$ and k controls the relative intensity of poloidal to toroidal magnetic field. In the present work we use $a = 127\rho_{s0}$, $k = 0.06$, $r_{\max} - r_{\min} = 90$ and $R_0 = 500\rho_{s0}$. We remark that, while ψ is not a solution of the Grad-Shafranov equation, it provides an analytical expression to compute the source term, $M(u) = S$.
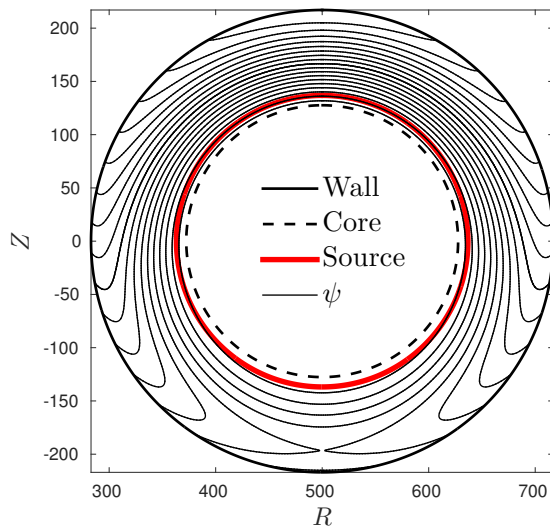

Figure 4: Contour lines of the flux function in eq (35) used for GBS code verification and convergence tests.

The manufactured solution for the evolved quantities $f = n, T_{e,i}, v_{\parallel e,i}, \phi, \omega$ are chosen to have the form

$$u(r, \theta, \varphi; t) = A_f[B_f + \sin(C_f\varphi)\sin(D_f\theta)\sin(E_f t + F_f r)]$$

where A_f, B_f, C_f, D_f, E_f and F_f are arbitrary constants that may be different for each field f and are tuned to excite all the terms in the model equations.

The source term $S = M(u)$ is computed by using Mathematica software package [33], and it is added to the GBS model equation. The results of the GBS verification confirms that $p \to 4$ for $h \to 0$ for both the L_∞ (fig. 5, left) and L_2 (fig. 5, right) norm. These results do not include the curvature parts of gyroviscous terms, as they are implemented at second order and have been verified independently. The boundary conditions are not considered in this study.
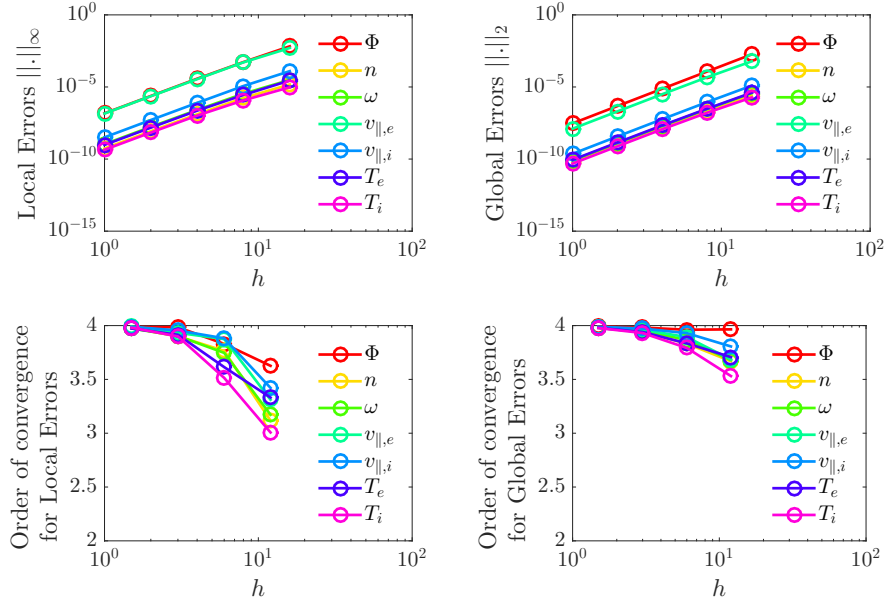
Figure 5: GBS code verification by the method of manufactured solution. The error of the numerical solution to the analytical manufactured one is shown as a function of the grid size h, both in L_∞ (top left) and L_2 norm (top right). The order of convergence p tends to 4 as h decreases for both norms (bottom left and right), consistent with the 4th order finite difference numerical scheme used.
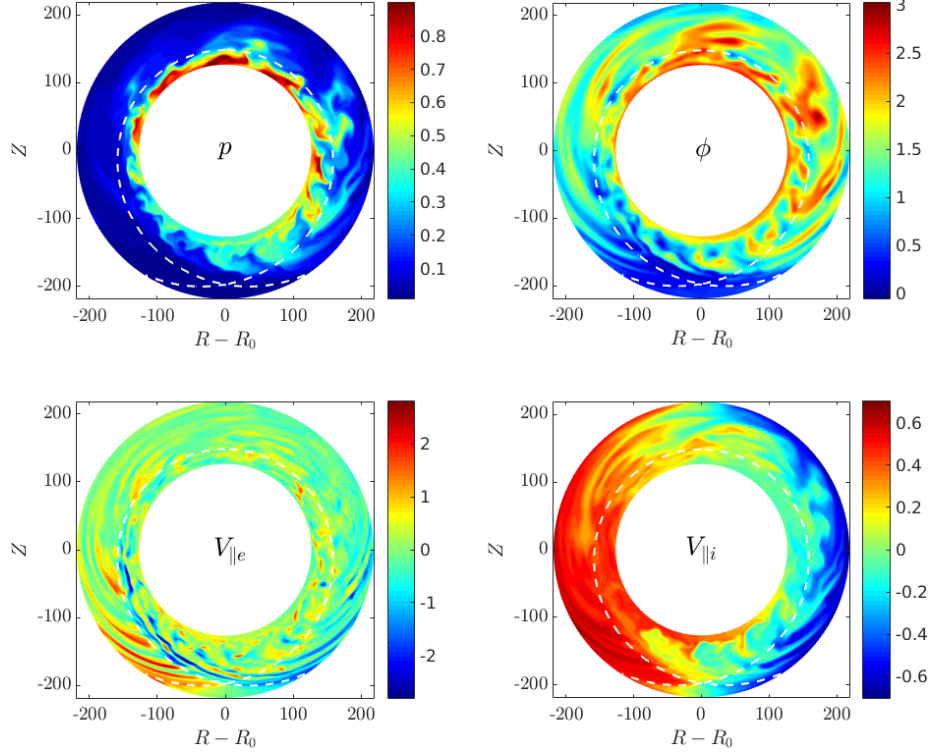
5   Convergence study

Figure 6: Typical snapshot in the poloidal plane of plasma pressure ($p = n(T_e + \tau T_i)$, normalised to $n_0 T_0$), electric potential, parallel electron and ion velocities for the analytical in flux function shown fig. 4. The plasma is mainly confined inside closed field line region, turbulent eddies are sheared at the separatrix (white dashed line) and form blob structures that move radially outwards and are eventually lost at the wall. The simulation with *fine* grid is considered.
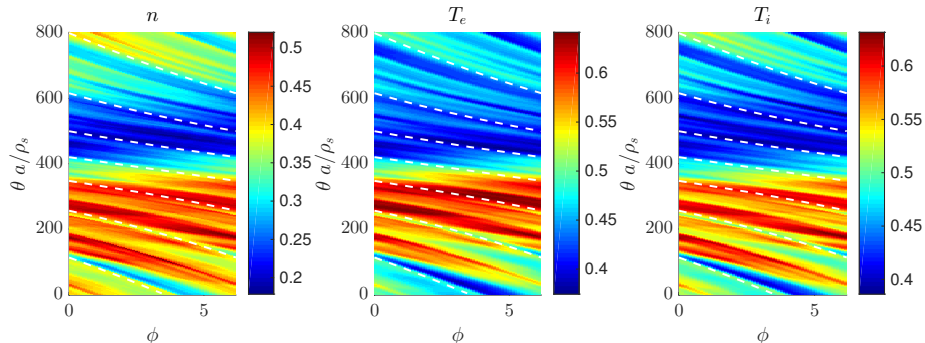
Figure 7: Typical snapshot of density and electron and ion temperatures at the $r = 20$ plane inside the separatrix. Turbulence structures follow the magnetic field lines, traced by white dashed lines. The same simulation and time frame of fig. 6 is considered.

The GBS convergence with respect to the grid refinement is tested with the flux function in eq. (35). Three simulations with increasing spatial grid resolution are compared: a *coarse* simulation with grid $N_r \times N_\theta \times N_\varphi = 39 \times 122 \times 16$, a *medium* simulation with grid $N_r \times N_\theta \times N_\varphi = 78 \times 244 \times 32$, and a *fine* simulation with grid $N_r \times N_\theta \times N_\varphi = 156 \times 488 \times 64$. The time step is chosen to grant stability. Typical snapshots from the *fine* simulation are reported in fig. 6 and 7 showing turbulence structures that are field aligned. We perform the convergence analysis focusing on time averaged profiles, obtained after the system has reached a quasi steady state. This sets in when the inflow of density and temperature due to the sources is balanced by parallel and radial losses at the wall, resulting in fluctuations around an approximately constant value.
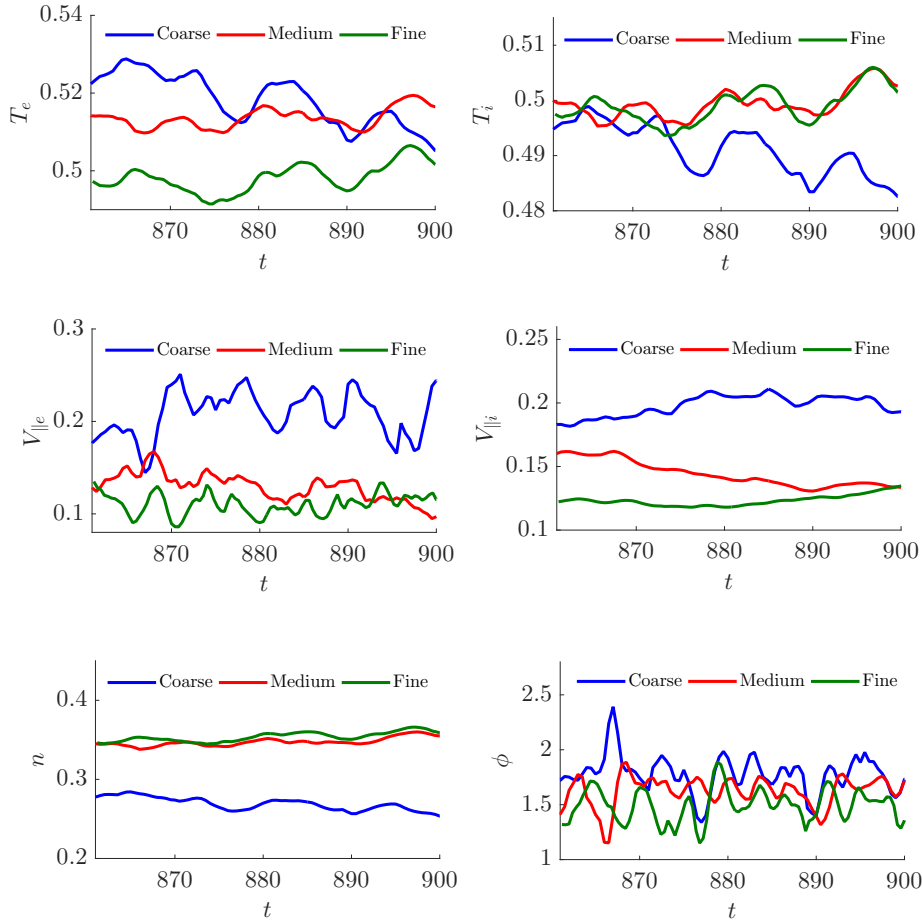
Figure 8: Time trace of $T_e, T_i, v_{\parallel i}, v_{\parallel e}, n$ and ϕ averaged over the 3D domain for simulation of increasing grid resolution at quasi-steady state: *coarse* grid $(N_r \times N_\theta \times N_\varphi = 39 \times 122 \times 16)$, *medium* grid $(N_r \times N_\theta \times N_\varphi = 78 \times 244 \times 32)$ and *fine* grid $(N_r \times N_\theta \times N_\varphi = 156 \times 488 \times 64)$.

Fig. 8 shows the averaged values of $T_e, T_i, v_{\parallel e}, v_{\parallel i}, \phi$ and n over the entire domain during quasi-steady state. The plot shows qualitatively the convergence of the code results with the grid resolution, inasmuch as the profiles of *fine* and *medium* are close to each other, while the *coarse* grid traces are slightly off. Convergence is evident for n, T_i, $v_{\parallel e}$ and $v_{\parallel i}$. For ϕ the three average values are close to each other, being in overall agreement. Finally, for the electron temperature T_e the trend displayed by the *fine* and *medium* simulations is similar, although the average values differ slightly, and the *coarse* T_e profile oscillates somewhere in between.
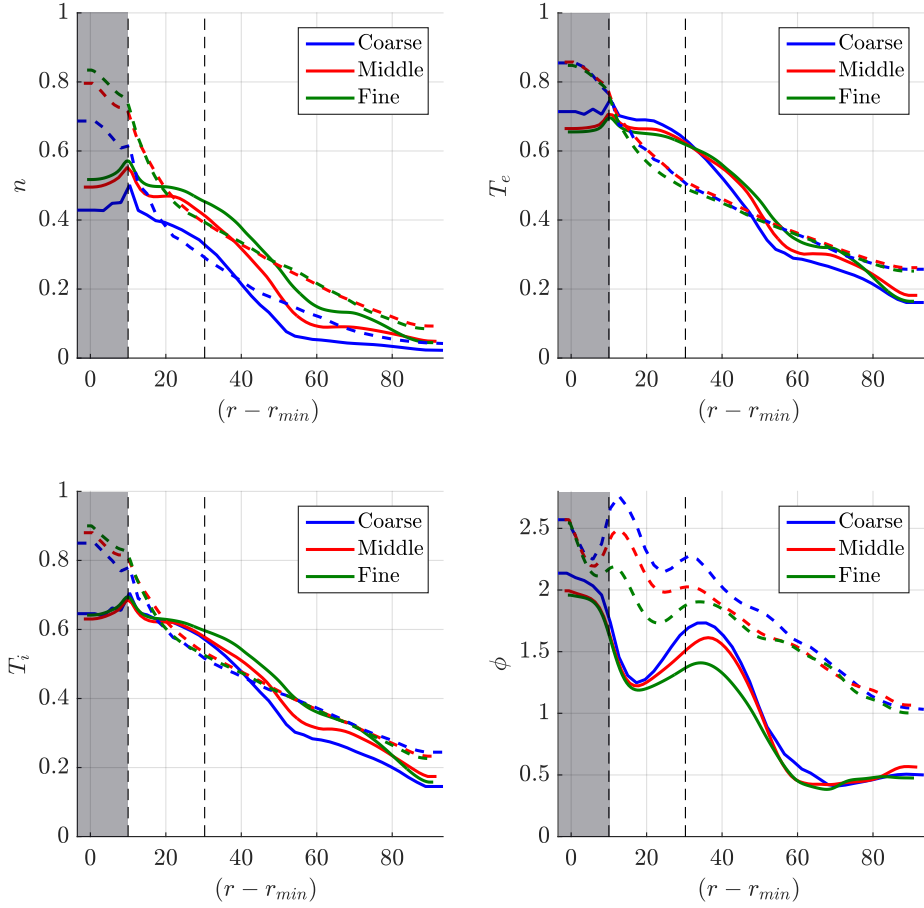
Figure 9: Toroidal and temporal average of radial profiles at the LFS (solid lines) and HFS (dashed lines) for n, T_e, T_i, ϕ resulting from GBS simulations carried out for the three different resolutions in fig. 8. The vertical lines at $r - r_{min} \sim 30$ show the radial position of the separatrix, while the shaded area is the buffer zone between the inner radial boundary and the plasma source position.

The toroidal and time averaged radial profiles of n, T_e, T_i and ϕ, which are often used to predict SOL width (see e.g. [9]), are shown in fig. 9 on the equatorial midplane at the low field side (LFS) and high field side (HFS), with a solid and dashed line, respectively. The vertical dashed line at $r - r_{min} \sim 30$ indicates the separatrix location. The shaded region that extends from the inner radial boundary to the source location is a buffer volume, which is not subject of physics investigations. The three simulations show qualitative agreement for all fields with clear convergence pattern for n, T_e, T_i.

To study the convergence of the $v_{\parallel e}$ and $v_{\parallel i}$ fields we analyse their time and

toroidally averaged profiles along the separatrix. Since we are not using flux coordinates, the values on the separatrix are obtained by performing a linear interpolation between the grid points. In fig. 10 the averaged values of $v_{\parallel e}$ and $v_{\parallel i}$ are plotted against s, a coordinate that maps the separatrix and it is normalised to ρ_{s0}. We impose $s = 0$ at the divertor plate at the HFS, the coordinate s increases moving along the inner divertor leg. The value of s at the X-point is indicated by the first vertical line. Larger values of s parametrise the loop around the separatrix from the HFS to the LFS until the X-point position (indicated by the second vertical line). Finally, s tracks the outer leg up to the wall. The results of the three simulations are again in good agreement and show convergence with the refinement of the grid, in particular for $v_{\parallel i}$.
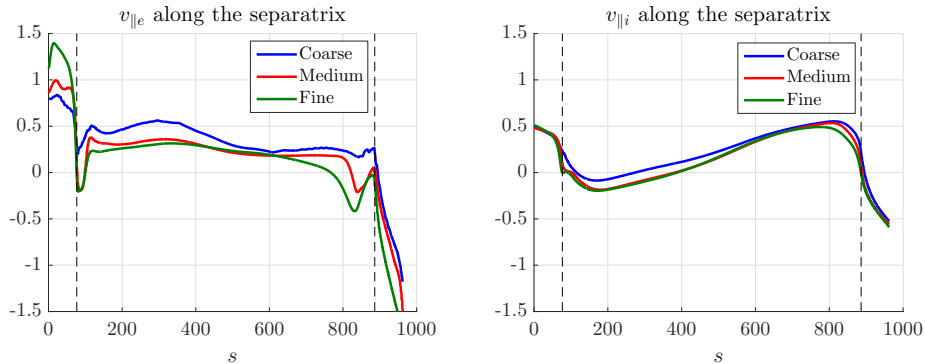

Figure 10: Toroidal and time average of the velocities profiles as a function of s, a coordinate that maps the separatrix from the HFS to the LFS, for the three different resolutions in fig. 8

To conclude the convergence analysis with a quantitative evaluation, we consider the time and toroidally averaged profiles of all fields in the (r, θ) poloidal plane. We use as index of convergence the distance, in the sense of the L_2 norm on the poloidal (r, θ) plane, between the *coarse* and the *fine* simulations and between the *medium* and the *fine* ones. This is represented in fig. 11, where h/h_0 indicates the ratio of the coarser grids to the refined grid spacing. The distance to the refined simulation is smaller for the *medium* grid than for the *coarse* grid for all fields, with an indicative order of convergence, evaluated from the slope of the lines in fig. 11 ranging from approximately 2 for ω and ϕ, to approximately 5 for n.
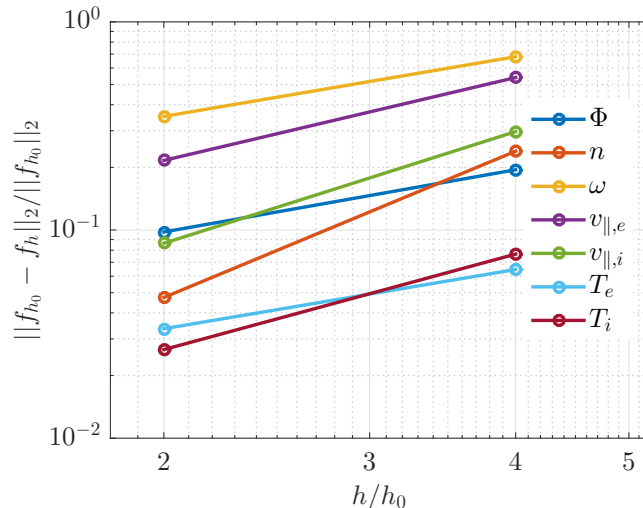
20

Figure 11: Quantitative estimate of code convergence. The toroidal and temporal average of $\phi, n, \omega, v_{\parallel e}, v_{\parallel i}, T_e, T_i$ for the *coarse* and *medium* grid is compared to the *fine* grid, by computing the L_2 norm in (r, θ) of their difference. The parameter h_0 denotes the grid spacing of the *fine* grid and h the grid spacing of the coarser grids, so that $h/h_0 = 4$ and $h/h_0 = 2$ for the *coarse* and *medium* grid, respectively.

6 Conclusion

In the present paper, a new version of GBS for the treatment of diverted equilibria is presented and successfully tested. With respect to the limited version of GBS, we use toroidal coordinates to express the differential operators present in the drift-reduced Braginskii's equations. Under the assumption that gradient length scales are of the same order of the major radius in the toroidal direction, and smaller than the minor radius in the poloidal plane, as well as by neglecting terms that are order one or higher in the inverse aspect-ratio, we can simplify the differential operators that act on the planal perpendicular to the magnetic field, discarding toroidal derivatives.

The coordinates are discretised by using a uniform grid. To compensate the lack of alignment between toroidal coordinates and magnetic field, the spatial accuracy of the numerical operators is increased from second to fourth order, and additional grid staggering is added in the poloidal direction.

The GBS code implementation is successfully verified through the MMS. Good convergence behaviour with the refining of the grid resolution is obtained for an analytical flux function describing a diverted configuration. By remaining relatively simple, the numerical scheme we present allows for an easy and efficient parallelisation of GBS. In addition it is flexible since equilibria with multiple X-points or snowflakes can be

easily implemented.

Two aspects of the implementation are, for the time being, somewhat arbitrary and require further work. These are the boundary conditions at the inner radial boundary, which describes the interaction between the core and the edge of the tokamak, and the plasma behaviour at the outer radial boundary, at the locations where the magnetic field lines are tangent to the wall and the magnetic pre-sheath boundary model implemented in GBS fails.

Finally, most magnetic equilibria used in experiments present a vertically elongated shape. As a consequence, the use of a circular ring as a computational domain results into an inefficient use of resources. In these cases the use of toroidal coordinates with a vertical elongation, such that constant r surfaces map ellipses instead of circles, is a relatively straightforward possible future implementation to exploit more efficiently the computational domain.

7 Acknowledgments

A Spatial differential operators in toroidal coordinates

Hereafter we detail the derivation procedure to obtain the simplified expressions of the five differential operators $\nabla_\|$, $\nabla_\|^2$, C, $[\phi, \ ]$, $\nabla_\perp^2$, appearing in the drift-reduced Braginskii's equations (3)-(9), in toroidal coordinates. The toroidal coordinates in physical units $(\tilde{r}, \theta, \varphi)$ read:

$$
\begin{aligned}
x &= \tilde{R}\cos\varphi = (R_0 - \tilde{r}\cos\theta)\cos\varphi \\
y &= \tilde{R}\sin\varphi = (R_0 - \tilde{r}\cos\theta)\sin\varphi \\
z &= Z_0 + \tilde{r}\sin\theta
\end{aligned}
\tag{36}
$$

and have an associated covariant coordinates basis [34],

$$
\begin{aligned}
\mathbf{e}_{\tilde{r}} &= -\cos\theta\cos\varphi\,\mathbf{e}_x - \cos\theta\sin\varphi\,\mathbf{e}_y + \sin\theta\,\mathbf{e}_z \\
\mathbf{e}_\theta &= \tilde{r}(\sin\theta\cos\varphi\,\mathbf{e}_x + \sin\theta\sin\varphi\,\mathbf{e}_y + \cos\theta\,\mathbf{e}_z) \\
\mathbf{e}_\varphi &= \tilde{R}(-\sin\varphi\,\mathbf{e}_x + \cos\varphi\,\mathbf{e}_y)
\end{aligned}
\tag{37}
$$

The contravariant basis [34] $(\mathbf{e}^{\tilde{r}}, \mathbf{e}^{\theta}, \mathbf{e}^{\varphi}) = (\nabla \tilde{r}, \nabla \theta, \nabla \varphi)$ is defined such that:

$$\nabla \tilde{r} = \frac{\mathbf{e}_{\tilde{r}}}{||\mathbf{e}_{\tilde{r}}||^2}, \qquad \nabla \theta = \frac{\mathbf{e}_{\theta}}{||\mathbf{e}_{\theta}||^2}, \qquad \nabla \varphi = \frac{\mathbf{e}_{\varphi}}{||\mathbf{e}_{\varphi}||^2}. \tag{38}$$

Finally the Jacobian of the transformation from (x, y, z) to $(\tilde{r}, \theta, \varphi)$ is $\tilde{J} = (\nabla \tilde{r} \cdot \nabla \theta \times \nabla \varphi)^{-1} = \tilde{r} \tilde{R}$.

We now express the axisymmetric tokamak magnetic field in the form:

$$\tilde{\mathbf{B}} = F(\tilde{\psi}) \nabla \varphi + \nabla \varphi \times \nabla \tilde{\psi}(\tilde{r}, \theta)$$

where, for simplicity, we assume $F = R_0 B_0$. Therefore, the expression of the magnetic field in covariant components, $\tilde{\mathbf{B}} = \tilde{B}_i \mathbf{e}^i$, reads as

$$\tilde{\mathbf{B}} = B_0 R_0 \ \nabla \varphi - \frac{1}{\tilde{r}\tilde{R}} \partial_\theta \tilde{\psi} \ \nabla \tilde{r} + \frac{\tilde{r}}{\tilde{R}} \partial_{\tilde{r}} \tilde{\psi} \ \nabla \theta \tag{39}$$

and in controvariant components, $\mathbf{B} = \tilde{B}^i \mathbf{e}_i$,

$$\tilde{\mathbf{B}} = \frac{B_0 R_0}{\tilde{R}^2} \ \mathbf{e}_{\varphi} - \frac{1}{\tilde{r}\tilde{R}} \partial_\theta \tilde{\psi} \ \mathbf{e}_{\tilde{r}} + \frac{1}{\tilde{r}\tilde{R}} \partial_{\tilde{r}} \tilde{\psi} \ \mathbf{e}_{\theta} \tag{40}$$

Eqs. (36)-(47) provide all the elements to evaluate the expressions of the differential operators in toroidal coordinates. In physical units these are (note the use of Einstein notation)

$$\begin{aligned}
\tilde{\nabla}_{\parallel} f &= \frac{\tilde{\mathbf{B}}}{\tilde{B}} \cdot \left(\nabla \tilde{r} \frac{\partial f}{\partial \tilde{r}} + \nabla \theta \frac{\partial f}{\partial \theta} + \nabla \varphi \frac{\partial f}{\partial \varphi} \right) \\
&= \frac{1}{\tilde{B}} \left(\tilde{B}^{\tilde{r}} \frac{\partial f}{\partial \tilde{r}} + \tilde{B}^{\theta} \frac{\partial f}{\partial \theta} + \tilde{B}^{\varphi} \frac{\partial f}{\partial \varphi} \right)
\end{aligned} \tag{41}$$

$$\begin{aligned}
[\tilde{\phi, f}] &= \frac{\tilde{\mathbf{B}}}{\tilde{B}} \cdot \left[\left(\frac{\partial \phi}{\partial \tilde{r}} \nabla \tilde{r} + \frac{\partial \phi}{\partial \theta} \nabla \theta + \frac{\partial \phi}{\partial \varphi} \nabla \varphi \right) \times \left(\frac{\partial f}{\partial \tilde{r}} \nabla \tilde{r} + \frac{\partial f}{\partial \theta} \nabla \theta + \frac{\partial f}{\partial \varphi} \nabla \varphi \right) \right] \\
&= \frac{1}{\tilde{B}\tilde{J}} \left(\tilde{B}_{\varphi} [\phi, f]_{\tilde{r},\theta} + \tilde{B}_{\tilde{r}} [\phi, f]_{\theta,\varphi} + \tilde{B}_{\theta} [\phi, f]_{\varphi,\tilde{r}} \right)
\end{aligned} \tag{42}$$

$$\begin{aligned}
\tilde{C}(f) &= \frac{1}{2\tilde{J}\tilde{B}} \Big[- \frac{\tilde{B}_{\varphi}}{\tilde{B}^2} \partial_\theta \tilde{B}^2 \frac{\partial f}{\partial \tilde{r}} + \frac{\tilde{B}_{\varphi}}{\tilde{B}^2} \partial_{\tilde{r}} \tilde{B}^2 \frac{\partial f}{\partial \theta} \\
&\quad + \left(\partial_{\tilde{r}} \tilde{B}_{\theta} - \partial_\theta \tilde{B}_{\tilde{r}} - \frac{\tilde{B}_{\theta}}{\tilde{B}^2} \partial_{\tilde{r}} \tilde{B}^2 + \frac{\tilde{B}_{\tilde{r}}}{\tilde{B}^2} \partial_\theta \tilde{B}^2 \right) \frac{\partial f}{\partial \varphi} \Big]
\end{aligned} \tag{43}$$

$$\begin{aligned}
\tilde{\nabla}_{\perp}^2 f &= \tilde{\nabla} \cdot \tilde{\nabla}_{\perp} f = \frac{1}{\tilde{J}} \frac{\partial}{\partial u^k} \left(\tilde{J} (\nabla_{\perp} f)^k \right) \\
&= \frac{1}{\tilde{J}} \sum_k \frac{\partial}{\partial u^k} \left(\frac{\tilde{J}}{\tilde{B}^2} ||\mathbf{e}^k||^2 \epsilon_{ijk} \epsilon^{lni} \tilde{B}_l \tilde{B}^j \frac{\partial f}{\partial u^n} \right)
\end{aligned} \tag{44}$$

where we define $[\phi, f]_{x,y} := \partial_x \phi \partial_y f - \partial_y \phi \partial_x f$

We note that the poloidal flux varies on the length scale of a, that the toroidal gradients of the evolved quantities f vary on the length scale of R_0, while their gradients in the poloidal plane vary on the length scale of l_p, with $\rho_{s0} < l_p < a$. We therefore simplify the expressions above by

retaining the leading order terms in $\varepsilon = a/R_0$, $\sigma = l_p/R_0$ and $\delta = \rho_{s0}/R_0$

Since all operators are written in terms of the covariant and controvariant components of the magnetic field, let us start by considering their ordering in terms of a, R_0 and B_0. For the covariant components in eq. (39) we observe:

$$\tilde{\mathbf{B}} = -\underbrace{\frac{1}{\tilde{r}\tilde{R}}}_{a^{-1}R_0^{-1}}\underbrace{\partial_\theta\tilde{\psi}}_{a^2 B_0}\underbrace{\nabla\tilde{r}}_{1} + \underbrace{\frac{\tilde{r}}{\tilde{R}}}_{aR_0^{-1}}\underbrace{\partial_{\tilde{r}}\tilde{\psi}}_{aB_0}\underbrace{\nabla\theta}_{a^{-1}} + \underbrace{B_0 R_0}_{R_0 B_0}\underbrace{\nabla\varphi}_{R_0^{-1}}$$

$$= \underbrace{\tilde{B}_{\tilde{r}}}_{aR_0^{-1}B_0}\nabla\tilde{r} + \underbrace{\tilde{B}_\theta}_{a^2 R_0^{-1}B_0}\nabla\theta + \underbrace{\tilde{B}_\varphi}_{R_0 B_0}\nabla\varphi \qquad (45)$$

where we assumed $\tilde{\psi} \sim a^2|B_0|$ and for the controvariant components in eq. (40):

$$\tilde{\mathbf{B}} = -\underbrace{\frac{1}{\tilde{r}\tilde{R}}}_{a^{-1}R_0^{-1}}\underbrace{\partial_\theta\tilde{\psi}}_{a^2 B_0}\underbrace{\mathbf{e}_{\tilde{r}}}_{1} + \underbrace{\frac{1}{\tilde{r}\tilde{R}}}_{a^{-1}R_0^{-1}}\underbrace{\partial_{\tilde{r}}\tilde{\psi}}_{aB_0}\underbrace{\mathbf{e}_\theta}_{a} + \underbrace{\frac{B_0 R_0}{\tilde{R}^2}}_{R_0^{-1}B_0}\underbrace{\mathbf{e}_\varphi}_{R_0}$$

$$= \underbrace{\tilde{B}^{\tilde{r}}}_{aR_0^{-1}B_0}\mathbf{e}_{\tilde{r}} + \underbrace{\tilde{B}^\theta}_{R_0^{-1}B_0}\mathbf{e}_\theta + \underbrace{\tilde{B}^\varphi}_{R_0^{-1}B_0}\mathbf{e}_\varphi \qquad (46)$$

The modulus of $\mathbf{B}$ is

$$\tilde{B}^2 = \tilde{B}_i\tilde{B}^i = \underbrace{\tilde{B}_\varphi\tilde{B}^\varphi}_{B_0^2} + \underbrace{\tilde{B}_{\tilde{r}}\tilde{B}^{\tilde{r}}}_{a^2 R_0^{-2}} + \underbrace{\tilde{B}_\theta\tilde{B}^\theta}_{a^2 R_0^{-2}} = B_0^2\frac{R_0^2}{\tilde{R}^2} + O(\varepsilon^2) \qquad (47)$$

Additionally it is useful to have the ordering for $\partial_{\tilde{r}}\tilde{B}^2$, $\partial_\theta\tilde{B}^2$, $\partial_{\tilde{r}}\tilde{B}_\theta$ and $\partial_\theta\tilde{B}_{\tilde{r}}$, that is

$$\partial_{\tilde{r}}\tilde{B}^2 = \frac{2}{\tilde{R}}\cos\theta\tilde{B}^2 + \left[-\frac{2}{\tilde{R}^2\tilde{r}^3}\partial_\theta\tilde{\psi}^2 + \frac{1}{\tilde{R}^2\tilde{r}^2}\partial_{\tilde{r}}(\partial_\theta\tilde{\psi})^2 + \frac{1}{\tilde{R}^2}\partial_{\tilde{r}}(\partial_{\tilde{r}}\tilde{\psi})^2\right] \sim \frac{B_0^2}{R_0} \qquad (48)$$

$$\partial_\theta\tilde{B}^2 = -\frac{2}{\tilde{R}}\tilde{r}\sin\theta\tilde{B}^2 + \left[\frac{1}{\tilde{R}^2\tilde{r}^2}\partial_\theta(\partial_\theta\tilde{\psi})^2 + \frac{1}{\tilde{R}^2}\partial_\theta(\partial_{\tilde{r}}\tilde{\psi})^2\right] \sim \frac{aB_0^2}{R_0} \qquad (49)$$

$$\partial_{\tilde{r}}\tilde{B}_\theta = \frac{1}{\tilde{R}}\partial_{\tilde{r}}\tilde{\psi} + \frac{\tilde{r}}{\tilde{R}}\partial^2_{\tilde{r}\tilde{r}}\tilde{\psi} + \left[\frac{\tilde{r}}{\tilde{R}^2}\cos\theta\partial_{\tilde{r}}\tilde{\psi}\right] \sim \frac{aB_0}{R_0} \qquad (50)$$

$$\partial_\theta\tilde{B}_{\tilde{r}} = -\frac{1}{\tilde{r}\tilde{R}}\partial^2_{\theta\theta}\tilde{\psi} - \left[\frac{\sin\theta}{\tilde{R}^2}\partial_\theta\tilde{\psi}\right] \sim \frac{aB_0}{R_0} \qquad (51)$$

With the terms inside the square brackets being order ε higher than the leading order terms.

We now have all the ingredients to write the dimensionless differential operators in toroidal coordinates, neglecting all terms that are order one or higher in ε, σ and δ. We show the full derivation of the curvature operator as an example for all the others. From eq. (43) and eq. (15) and using the ordering for the magnetic field and its derivatives in eqs. (45)-(51) we

obtain the following dimensionless form of the curvature operator:

$$
C(f) = - \overbrace{\underbrace{\frac{\rho_{s0} R_0}{2\tilde{J}}}_{\rho_{s0} a^{-1}} \underbrace{\frac{\tilde{B}_\varphi}{\tilde{B}}}_{R_0} \underbrace{\frac{\partial_\theta \tilde{B}^2}{\tilde{B}^2}}_{aR_0^{-1}} \underbrace{\frac{\partial f}{\partial \tilde{r}}}_{l_p^{-1}}}^{\delta\sigma^{-1}} + \overbrace{\underbrace{\frac{\rho_{s0} R_0}{2\tilde{J}}}_{\rho_{s0} a^{-1}} \underbrace{\frac{\tilde{B}_\varphi}{\tilde{B}}}_{R_0} \underbrace{\frac{\partial_{\tilde{r}} \tilde{B}^2}{\tilde{B}^2}}_{R_0^{-1}} \underbrace{\frac{\partial f}{\partial \theta}}_{l_p^{-1} a}}^{\delta\sigma^{-1}}
$$

$$
+ \overbrace{\underbrace{\frac{\rho_{s0} R_0}{2\tilde{J}}}_{\rho_{s0} a^{-1}} \left(\underbrace{\frac{\partial_{\tilde{r}} \tilde{B}_\theta}{\tilde{B}}}_{aR_0^{-1}} - \underbrace{\frac{\partial_\theta \tilde{B}_{\tilde{r}}}{B}}_{aR_0^{-1}} - \underbrace{\frac{\tilde{B}_\theta}{\tilde{B}}}_{a^2 R_0^{-1}} \underbrace{\frac{\partial_{\tilde{r}} \tilde{B}^2}{\tilde{B}^2}}_{R_0^{-1}} + \underbrace{\frac{\tilde{B}_{\tilde{r}}}{\tilde{B}}}_{aR_0^{-1}} \underbrace{\frac{\partial_\theta \tilde{B}^2}{\tilde{B}^2}}_{aR_0^{-1}} \right) \underbrace{\frac{\partial f}{\partial \varphi}}_{1}}^{\delta}
$$

The first two terms containing the poloidal derivatives of f are order $\delta\sigma^{-1} = \rho_{s0}/l_p$ and are non-negligible, since $a > l_p > \rho_{s0}$ and $\varepsilon < \delta\sigma^{-1} < 1$. The third term containing the toroidal derivative is instead order $\delta \sim \rho_{s0}/R_0$ and can be neglected. Substituting the expression for $\tilde{B}$, $\tilde{B}_\varphi$, $\partial_\theta \tilde{B}^2$, $\partial_{\tilde{r}} \tilde{B}^2$ and $\tilde{J}$ we obtain the final expression for $C(f)$ implemented in GBS reported in eq. (20):

$$
C(f) = \frac{B_0}{\tilde{B}} \frac{R_0^2}{\tilde{R}^2} \left(\sin\theta \frac{\partial f}{\partial r} + \frac{\cos\theta}{r} \frac{\partial f}{\partial \theta} \right) + O(\delta)
$$

$$
= \frac{B_0}{|B_0|} \left(\sin\theta \frac{\partial f}{\partial r} + \frac{\cos\theta}{r} \frac{\partial f}{\partial \theta} \right) + O(\varepsilon\delta\sigma^{-1}) \tag{52}
$$

In eq. (52) we used the dimensionless radial coordinate $r = \tilde{r}/\rho_{s0}$ as well as $\tilde{R} = R_0(1 + O(\varepsilon))$ and $\tilde{B} = |B_0| + O(\varepsilon)$, according to eqs. (36) and (47). Note that $\varepsilon\delta\sigma^{-1} < \varepsilon$.

References

[1] A. Loarte, B. Lipschultz, A. Kukushkin, G. Matthews, P. Stangeby, N. Asakura, G. Counsell, G. Federici, A. Kallenbach, K. Krieger, *et al.*, "Power and particle control," *Nuclear Fusion*, vol. 47, no. 6, p. S203, 2007.

[2] B. Dudson, A. Allen, G. Breyiannis, E. Brugger, J. Buchanan, L. Easy, S. Farley, I. Joseph, M. Kim, A. McGann, *et al.*, "Bout++: Recent and current developments," *Journal of Plasma Physics*, vol. 81, no. 1, 2015.

[3] P. Tamain, H. Bufferand, G. Ciraolo, C. Colin, D. Galassi, P. Ghendrih, F. Schwander, and E. Serre, "The tokam3x code for edge turbulence fluid simulations of tokamak plasmas in versatile magnetic geometries," *J. Comput. Phys.*, vol. 321, pp. 606–623, Sept. 2016.

[4] B. Zhu, M. Francisquez, and B. N. Rogers, "Global 3d two-fluid simulations of the tokamak edge region: Turbulence, transport, profile evolution, and spontaneous e× b rotation," *Physics of Plasmas*, vol. 24, no. 5, p. 055903, 2017.

[5] A. Stegmeir, D. P. Coster, A. Ross, O. Maj, K. Lackner, and E. Poli, "Grillix: A 3d turbulence code based on the flux-coordinate independent approach," *Plasma Physics and Controlled Fusion*, 2017.

[6] A. H. Nielsen, G. Xu, J. Madsen, V. Naulin, J. J. Rasmussen, and B. Wan, "Simulation of transition dynamics to high confinement in fusion plasmas," *Physics Letters A*, vol. 379, no. 47, pp. 3097–3101, 2015.

[7] S. I. Braginskii, "Transport Processes in a Plasma," *Reviews of Plasma Physics*, vol. 1, p. 205, 1965.

[8] A. Zeiler, J. F. Drake, and B. Rogers, "Nonlinear reduced braginskii equations with ion thermal dynamics in toroidal plasma," *Physics of Plasmas*, vol. 4, no. 6, pp. 2134–2138, 1997.

[9] F. D. Halpern, P. Ricci, B. Labit, I. Furno, S. Jolliet, J. Loizu, A. Mosetto, G. Arnoux, J. Gunn, J. Horacek, *et al.*, "Theory-based scaling of the sol width in circular limited tokamak plasmas," *Nuclear Fusion*, vol. 53, no. 12, p. 122001, 2013.

[10] F. Riva, E. Lanti, S. Jolliet, and P. Ricci, "Plasma shaping effects on tokamak scrape-off layer turbulence," *Plasma Physics and Controlled Fusion*, vol. 59, no. 3, p. 035001, 2017.

[11] J. P. Freidberg, *Ideal MHD.* Cambridge University Press, 2014.

[12] F. Hariri and M. Ottaviani, "A flux-coordinate independent field-aligned approach to plasma turbulence simulations," *Computer Physics Communications*, vol. 184, no. 11, pp. 2419–2429, 2013.

[13] A. Stegmeir, D. Coster, O. Maj, K. Hallatschek, and K. Lackner, "The field line map approach for simulations of magnetically confined plasmas," *Computer Physics Communications*, vol. 198, pp. 139–153, 2016.

[14] B. LaBombard, E. Marmar, J. Irby, J. Terry, R. Vieira, G. Wallace, D. Whyte, S. Wolfe, S. Wukitch, S. Baek, *et al.*, "Adx: a high field, high power density, advanced divertor and rf tokamak," *Nuclear Fusion*, vol. 55, no. 5, p. 053020, 2015.

[15] R. Ambrosino, R. Albanese, S. Coda, M. Mattei, J.-M. Moret, and H. Reimerdes, "Optimization of experimental snowflake configurations on tcv," *Nuclear Fusion*, vol. 54, no. 12, p. 123008, 2014.

[16] P. Ricci, F. Halpern, S. Jolliet, J. Loizu, A. Mosetto, A. Fasoli, I. Furno, and C. Theiler, "Simulation of plasma turbulence in scrape-off layer conditions: the gbs code, simulation results and code validation," *Plasma Physics and Controlled Fusion*, vol. 54, no. 12, p. 124047, 2012.

[17] A. Mosetto, F. D. Halpern, S. Jolliet, J. Loizu, and P. Ricci, "Finite ion temperature effects on scrape-off layer turbulence," *Physics of Plasmas*, vol. 22, no. 1, p. 012308, 2015.

[18] F. D. Halpern, S. Jolliet, J. Loizu, A. Mosetto, and P. Ricci, "Ideal ballooning modes in the tokamak scrape-off layer," *Physics of Plasmas*, vol. 20, no. 5, p. 052306, 2013.

[19] C. Wersal and P. Ricci, "A first-principles self-consistent model of plasma turbulence and kinetic neutral dynamics in the tokamak scrape-off layer," *Nuclear Fusion*, vol. 55, no. 12, p. 123014, 2015.

[20] F. Halpern, P. Ricci, S. Jolliet, J. Loizu, J. Morales, A. Mosetto, F. Musil, F. Riva, T. Tran, and C. Wersal, "The {GBS} code for tokamak scrape-off layer simulations," *Journal of Computational Physics*, vol. 315, pp. 388 – 408, 2016.

[21] J. Loizu, P. Ricci, F. D. Halpern, and S. Jolliet, "Boundary conditions for plasma fluid models at the magnetic presheath entrance," *Physics of Plasmas*, vol. 19, no. 12, p. 122307, 2012.

[22] F. H. Harlow and J. E. Welch, "Numerical calculation of timedependent viscous incompressible flow of fluid with free surface," *The Physics of Fluids*, vol. 8, no. 12, pp. 2182–2189, 1965.

[23] S. Patankar, *Numerical heat transfer and fluid flow*. CRC press, 1980.

[24] S. Jolliet, F. Halpern, J. Loizu, A. Mosetto, F. Riva, and P. Ricci, "Numerical approach to the parallel gradient operator in tokamak scrape-off layer turbulence simulations and application to the gbs code," *Computer Physics Communications*, vol. 188, pp. 21–32, 2015.

[25] A. Arakawa, "Computational design for long-term numerical integration of the equations of fluid motion: Two-dimensional incompressible flow. part i," *Journal of Computational Physics*, vol. 1, no. 1, pp. 119 – 143, 1966.

[26] J. L. Peterson and G. W. Hammett, "Positivity preservation and advection algorithms with applications to edge plasma turbulence," *SIAM Journal on Scientific Computing*, vol. 35, no. 3, pp. B576–B605, 2013.

[27] J. Moritz, E. Faudot, S. Devaux, and S. Heuraux, "The plasma-wall transition layers in the presence of collisions with a magnetic field parallel to the wall," *Physics of Plasmas*, vol. 25, no. 1, p. 013534, 2018.

[28] D. Coulette and G. Manfredi, "Kinetic simulations of the chodura and debye sheaths for magnetic fields with grazing incidence," *Plasma Physics and Controlled Fusion*, vol. 58, no. 2, p. 025008, 2016.

[29] P. R. Amestoy, I. S. Duff, J. Koster, and J.-Y. L'Excellent, "A fully asynchronous multifrontal solver using distributed dynamic scheduling," *SIAM Journal on Matrix Analysis and Applications*, vol. 23, no. 1, pp. 15–41, 2001.

[30] P. R. Amestoy, A. Guermouche, J.-Y. L'Excellent, and S. Pralet, "Hybrid scheduling for the parallel solution of linear systems," *Parallel Computing*, vol. 32, no. 2, pp. 136–156, 2006.

[31] P. J. Roache, "Code verification by the method of manufactured solutions," *Journal of Fluids Engineering*, vol. 124, no. 1, pp. 4–10, 2002.

[32] F. Riva, P. Ricci, F. D. Halpern, S. Jolliet, J. Loizu, and A. Mosetto, "Verification methodology for plasma simulations and application to a scrape-off layer turbulence code," *Physics of Plasmas*, vol. 21, no. 6, p. 062301, 2014.

[33] I. Wolfram Research, "Mathematica," 2015.

[34] W. D. D'haeseleer, W. N. Hitchon, J. D. Callen, and J. L. Shohet, *Flux coordinates and magnetic field structure: a guide to a fundamental tool of plasma theory*. Springer Science & Business Media, 2012.